**Theorie der Informatik**

M. Helmert, G. Röger
F. Pommerening
Frühjahrssemester 2015

Universität Basel
Fachbereich Informatik

# Übungsblatt 12

**Abgabe: Mittwoch, 20. Mai 2015**

*Anmerkung:* Für Abgaben, die ausschliesslich mit LaTeX erstellt wurden, gibt es einen Bonuspunkt. Bitte geben Sie nur die resultierende PDF-Datei bzw. einen Ausdruck davon ab.

*Hinweis:* Dieses Übungblatt ist das letzte für die Vorlesung. Es gibt damit insgesamt ohne Bonuspunkte ein Maximum von 116 Punkten. Für die Prüfungszulassung werden wie angekündigt 50% dieser Punkte, also 58 Punkte, benötigt.

**Aufgabe 12.1** (Algorithmen für SAT, 1.5+1.5 Punkte + 1 Bonuspunkt)

Das *Erfüllbarkeitsproblem in der Aussagenlogik* (SAT) ist wie folgt definiert: gegeben eine aussagenlogische Formel $\varphi$, ist $\varphi$ erfüllbar?

(a) Geben Sie einen nichtdeterministischen Algorithmus für SAT an, dessen Laufzeit durch ein Polynom in der Länge von $\varphi$ beschränkt ist. Begründen Sie, warum die Laufzeit des Algorithmus polynomiell ist.

(b) Geben Sie einen deterministischen Algorithmus für SAT an.

(c) *Bonusaufgabe:* Schätzen Sie die Laufzeit Ihres Algorithmus aus Teil (b) in $O$-Notation ab.

Sie dürfen bei Ihrer Antwort beliebige übliche Programmierkonzepte verwenden. Es ist *nicht* nötig, dass Sie sich auf die eingeschränkte Syntax von WHILE-Programmen o.ä. beschränken. Es ist ausreichend high-level Pseudocode anzugeben, solange klar ist, dass jeder einzelne Schritt in polynomieller Zeit ausgeführt werden kann. Verwenden Sie die in der Vorlesung beschriebenen **GUESS**-Anweisungen für nichtdeterministische Anweisungen.

**Aufgabe 12.2** (P vs. NP, 1+1+1.5+1.5 Punkte)

Beweisen oder widerlegen Sie die folgenden Aussagen. Geben Sie in jedem Fall einen kurzen Beweis an (2–3 Sätze genügen).

(a) Sei $X$ ein NP-hartes Problem und $Y$ ein Problem, für das $X \leq_{\mathrm{p}} Y$ gilt. Dann ist $Y$ NP-hart.

(b) Sei $X$ ein NP-hartes Problem. Wenn für $X$ ein deterministischer polynomieller Algorithmus existiert, dann existiert für DirHamiltonCycle ein polynomieller Algorithmus.

(c) Es gibt NP-vollständige Probleme $X$ und $Y$, so dass für $X$ ein deterministischer polynomieller Algorithmus existiert, aber nicht für $Y$.

(d) Sei $Y \subseteq \Sigma^*$ ein beliebiges Problem mit $Y \neq \emptyset$ und $Y \neq \Sigma^*$. Für alle $X \in \mathrm{P}$ gilt $X \leq_{\mathrm{p}} Y$.

**Aufgabe 12.3** (Polynomielle Reduktion, 2 Punkte + 1 Bonuspunkt)

Ein *Hamiltonpfad* ist analog zu einem Hamiltonkreis definiert (vgl. Abschnitt 19.3), nur dass ein einfacher Pfad statt einem Kreis gesucht ist. Genauer: ein Hamiltonpfad in einem gerichteten Graphen $\langle V, E\rangle$ ist eine Knotenfolge $\pi = \langle v_1, \dots, v_n\rangle$, die einen Pfad definiert ($\langle v_i, v_{i+1}\rangle \in E$ für alle $1 \leq i < n$) und jeden Knoten des Graphen genau einmal enthält.
Betrachten Sie das Entscheidungsproblem $\textsc{DirHamiltonPathWithEndPoints}$:

- *Gegeben:* gerichteter Graph $G = \langle V, E\rangle$, Startknoten $v_\mathrm{s} \in V$, Endknoten $v_\mathrm{e} \in V$
- *Gefragt:* Enthält $G$ einen Hamiltonpfad von $v_\mathrm{s}$ nach $v_\mathrm{e}$, also einen Hamiltonpfad $\pi = \langle v_1, \dots, v_n\rangle$ mit $v_1 = v_\mathrm{s}$ und $v_n = v_\mathrm{e}$?

(a) Beweisen Sie, dass $\textsc{DirHamiltonPathWithEndPoints}$ NP-hart ist. Sie dürfen dabei verwenden, dass das Problem $\textsc{DirHamiltonCycle}$ NP-vollständig ist.

(b) *Bonusaufgabe:* Ist $\textsc{DirHamiltonPathWithEndPoints}$ NP-vollständig? Begründen Sie ihre Antwort.